ELICODE at MultiGED2023: fine-tuning XLM-RoBERTa for multilingual grammatical error detection

Davide Colla and **Matteo Delsanto** and **Elisa Di Nuovo**
University of Turin - Italy
Computer Science Department
davide.colla@unito.it, matteo.delsanto@unito.it, elisa.dinuovo@unito.it

Abstract

In this paper we describe the participation of our team, ELICODE, to the first shared task on Multilingual Grammatical Error Detection, MultiGED, organised within the workshop series on Natural Language Processing for Computer-Assisted Language Learning (NLP4CALL). The multilingual shared task includes five languages: Czech, English, German, Italian and Swedish. The shared task is tackled as a binary classification task at token level aiming at identifying correct or incorrect tokens in the provided sentences. The submitted system is a token classifier based on XLM-RoBERTa language model. We fine-tuned five different models—one per each language in the shared task. We devised two different experimental settings: first, we trained the models only on the provided training set, using the development set to select the model achieving the best performance across the training epochs; second, we trained each model jointly on training and development sets for 10 epochs, retaining the 10-epoch fine-tuned model. Our submitted systems, evaluated using F0.5 score, achieved the best performance in all evaluated test sets, except for the English REALEC data set (second classified). Code and models are publicly available at https://github.com/davidecolla/EliCoDe.

1 Introduction

Grammatical Error Detection (GED) is the task of automatically identifying errors in learner language. Despite its name, the errors to be identified are not only grammatical errors, but different error types are considered, e.g. spelling, punctuation, lexical. In Second Language Acquisition and Learner Corpus Research, indeed, an error is defined as "a linguistic form or combination of forms which, in the same context and under similar conditions of production, would, in all likelihood, not be produced by the speakers' native speaker counterparts" (Lennon, 1991). As can be noticed, this definition includes different causes, i.e. grammaticality and correctness, or acceptability, strangeness and infelicity (James, 1998). This difference results in different resources annotating different errors, with some annotating as grammatical errors also appropriateness errors—i.e. pragmatics, register and stylistic choices (Lüdeling and Hirschmann, 2015, p. 140)—others excluding appropriateness, but including orthographical and semantic well-formedness together with acceptability (Di Nuovo, 2022).

In both GED task and the related Grammatical Error Correction (GEC) task, research has focused mainly on learner English (as second or foreign language) (Bell et al., 2019; Ng et al., 2014; Bryant et al., 2019). Recently, also non-English error-annotated data sets have been released (Boyd, 2018; Náplava et al., 2022). Thanks to these recent trends, the authors of MultiGED (Volodina et al., 2023) organised this year the first multilingual GED shared task, hosted at the workshop series on Natural Language Processing for Computer-Assisted Language Learning (NLP4CALL).

Both GED and GEC can be seen as low or mid-resource tasks, because of three main characteristics: requiring time-expensive and highly-specialised human annotation, annotated data sets are usually small in size; the incorrect tokens in a text are significantly scarce if compared to the correct ones; since errors pertain to different error categories, each error type in the data sets is represented unevenly.

The data sets included in MultiGED shared task are in Czech, English, German, Italian and Swedish. Some of these data sets have been al-

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by/4.0/.

Davide Colla, Matteo Delsanto and Elisa Di Nuovo. EliCoDe at MultiGED2023: fine-tuning XLM-RoBERTa for multilingual grammatical error detection. *Proceedings of the 12th Workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL 2023)*. Linköping Electronic Conference Proceedings 197: 24–34.

ready used for GED or GEC tasks—i.e. *Falko* and *Merlin* corpora (Boyd, 2018), *Grammar Error Correction Corpus for Czech* (GECCC) (Náplava et al., 2022), *First Certificate in English* (FCE) corpus (Yannakoudakis et al., 2011)—others have been released *ad hoc* for this shared task—i.e. *Russian Error-Annotated Learner English Corpus* (REALEC) (Kuzmenko and Kutuzov, 2014), released only as development and test data sets, and learner Swedish *SweLL-gold* (Volodina et al., 2019), comprising training, development and test data sets.

The aim of MultiGED is to detect tokens to be corrected labelling them as correct or incorrect, performing a binary classification task at token level. Training and development data sets were segmented into sentences and tokens (no information at text level was released).

Following previous GED shared tasks, the used evaluation metric is F0.5, which weights precision twice as much as recall, carried out on the Codalab competition platform.[1]

The authors of the shared task encouraged submissions using a multilingual approach and additional resources, provided that these resources are publicly available for research purposes. However, since different resources can annotate different errors, the use of other additional data might be a double-edged sword. In fact, the additional data would increase the tool's ability to identify a greater variety of errors, but at the same time, as the tool is evaluated in-domain, it moves away from the characteristics of the test set.

In this paper, we present the systems submitted by our team, ELICODE, to MultiGED 2023 shared task. Our systems are both based on XLM-RoBERTa language model (Conneau et al., 2019), and do not use additional resources. We fine-tuned five models—one per each language in the shared task—for ten epochs. We devised two different experimental settings both using early stopping: in the first experimental setting, we trained the models only on the training data set and used the early stopping according to the F0.5 score obtained on the development data set (ELICODE); in the second experimental setting, we trained each model on both training and development data sets (ELICODE$_{ALL}$). Since in both experimental settings the early stopping was based on the development data set, in the second one, being it part of training, the training continued for all the ten epochs. We comment the results of the above-mentioned systems comparing them with a baseline—a Naive Bayes model—and an XLM-RoBERTa-based model trained jointly on the five-language training data sets (ELICODE$_{MLT}$) and on both training and development data sets (ELICODE$_{MLT_{ALL}}$), tackling the shared task with a multilingual approach.

The remainder of this paper is organised as follows: in Section 2 we present related work; in Section 3 we quantitatively describe the multilingual data set; in Section 4 we describe in detail our submitted models; in Section 5 we report and discuss the obtained results; in Section 6 we conclude the paper highlighting possible future work.

2 Related work

The detection of errors in interlanguage texts (Selinker, 1972) is a challenging task that has received significant attention in the natural language processing community, since GED systems can be used to provide feedback and guidance to language learners. In this section, we review some of the most relevant and recent studies in this area and in the related task of GEC.

Initially tackled using rule-based approaches, GED systems have evolved from being able to identify only certain types of errors to being more and more able to handle the complexity and variability of natural language, thanks to modern machine learning techniques which make use of large annotated text corpora, usually released in the occasion of shared tasks. This switch is evident in the evolution of the shared task from CoNLL-2013 (Ng et al., 2013) to CoNLL-2014 (Ng et al., 2014), when it changed from annotating only five error types to *all* error types.[2]

In CoNLL-2014 shared task, the majority of the systems made use of hybrid approaches able to deal with all error types together, as compared to previous year's submissions, where a specific classifier per each error type was trained. The most popular approaches made use of one or more of

[1] https://codalab.lisn.upsaclay.fr/competitions/9784

[2] Twenty-eight error types are annotated in the CoNLL-2014 benchmark data set. However, it should be noticed that this is still far from annotating all error types. For example, in the English Corpus of Learner English (ICLE) (Granger et al., 2020) there are 54 error tags, in the error-annotated learner Italian corpus, VALICO-UD (Di Nuovo, 2022, p. 94), 120 error tags.

the following: the Language Model (LM) based approach (using n-gram language models), which has been used for both GED and GEC; the phrase-based Statistical Machine Translation (SMT) approach, used mainly for GEC; and rule-based approaches to tackle regular error types.

In 2019, the Building Educational Applications (BEA) shared task on GEC (Bryant et al., 2019) introduces a new data set, joining the Cambridge English Write & Improve (W&I) (Yannakoudakis et al., 2018) and LOCNESS corpus (Granger, 1998), making the test data set bigger than the one on which CoNLL-2014 systems were tested (from 50 essays on two different topics, to 350 essays on about 50 topics). Another major change concerns the use of neural machine translation (Bryant et al., 2022)—being it based on recurrent neural networks (Bahdanau et al., 2014), convolutional neural networks (Gehring et al., 2016), or transformers (Vaswani et al., 2017)—instead of SMT and n-gram-based LMs. BEA reported results highlighted that the same system had different performances in texts at different CEFR levels (Little, 2006), lexical errors were the most difficult to detect and correct, and multi-token errors were better handled than in the previous shared task.

Bell et al. (2019) integrate contextual embeddings—BERT, ELMo and Flair embeddings (Peters et al., 2017; Devlin et al., 2018; Akbik et al., 2018)—in Rei (2017) architecture for GED (a bi-LSTM sequence labeler at token and sentence level, making use also of character-level bi-LSTM, to benefit from morphological information). Their best model used BERT embeddings and proved to better generalise in out-of-domain texts. Their analyses show that missing tokens are the most difficult errors to indentify.

Kaneko and Komachi (2019) proposed an extension of BERT base (Devlin et al., 2018) with multi-head multi-layer attention, since research has shown that different layers are best-suited for different tasks, e.g. lower layers capture local syntactic relationships, higher layers longer-range relationships (Peters et al., 2018).

Recently, Yuan et al. (2021) fine-tuned BERT, XLNet (Yang et al., 2019) and ELECTRA (Clark et al., 2020) models to perform GED in English. The three models obtained the new state of the art in binary GED training on FCE data set and testing on BEA-dev, FCE-test and CoNLL-2014, with ELECTRA performing the best overall. Thus, they used ELECTRA to carry out some multi-class GED experiments to boost performance on GEC data sets using it as auxiliary input or for re-ranking.

Our system treats GED as a binary sequence labelling task, like all the above-described systems, and since the best results have been obtained by fine-tuning transformer-based models, we followed this approach by fine-tuning XLM-RoBERTa model (Conneau et al., 2019). We decided to use multilingual RoBERTa because its training focuses on the discrimination of the masked token, and thus, it is conceptually similar to GED. In the following section we quantitatively analyse MultiGED data set, before describing in detail our submitted systems in Section 4.

3 Data set quantitative analysis

MultiGED data set contains labelled training and development sets in Czech (GECCC), English (FCE), Italian (Merlin), German (Falko and Merlin) and Swedish (SweLL-gold). In particular, for English language an additional data set (REALEC) has been released only as development set. In addition, for each data set an unlabelled test set has been released.

Following the work of Siino et al. (2022), we quantitatively analyse the 5-language data sets using established corpus linguistics methods implemented in Sketch Engine (Kilgarriff et al., 2014).[3] We report general data set figures in Table 1, as computed using Sketch Engine.

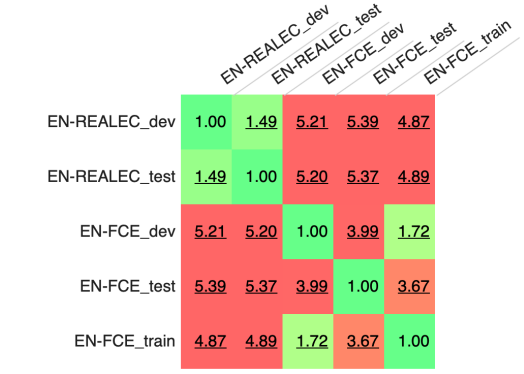
We used Compare Corpora, the built-in function of Sketch Engine that applies chi-square (χ^2) test (Kilgarriff, 2001), to compare training, development and test sets per each language. The result of this comparison is a confusion matrix per each language, reported in Figure 1, showing values greater or equal to 1, with 1 indicating identity. The higher the value, the larger the difference between the compared data sets.[4] For English we created a comprehensive confusion matrix comparing the two different corpora (FCE and REALEC).

[3] Available here: https://www.sketchengine.eu (last accessed on 28 March 2023).

[4] Please consider that correct or incorrect labels are not taken into account in this comparison. This comparison, instead, gives as an idea of how different the data sets are according to the different words used. Compare Corpora tool is affected by set size: this is why development and test sets, being the smallest, have a higher similarity score than when compared individually to the bigger training sets.

Source corpus	Language	Split	# Tokens	# Unique words
GECCC	Czech	train	333,995	37,228
		dev	32,071	8,145
		test	35,075	8,764
FCE	English	train	465,038	13,972
		dev	35,463	3,569
		test	42,545	3,800
REALEC	English	train	–	–
		dev	88,698	6,208
		test	90,391	6,300
Falko-MERLIN	German	train	306,847	20,561
		dev	39,627	5,606
		test	36,763	5,478
MERLIN	Italian	train	82,040	6,957
		dev	9,326	2,041
		test	10,300	2,176
SweLL-gold	Swedish	train	115,547	10,791
		dev	15,713	3,225
		test	14,666	3,141

Table 1: MultiGED data set in figures. # stands for *number of*.



(a) EN data sets.

(b) CS data set.

(c) DE data set.

(d) IT data set.

(e) SV data set.

Figure 1: Confusion matrices obtained with word-based chi-square test. The value 1.00 indicates identity between the compared data sets. The greater the value, the more different the data sets.

Looking at the matrices, we could suppose that systems should have less trouble in handling the task in German, Czech, Swedish (in order) than in Italian and English.

English (EN) data set – Since the big difference between FCE and REALEC, the lowest results should be obtained using models trained on FCE and tested on REALEC. Better results could be instead obtained fine-tuning in-domain using REALEC development set and testing it on the test set (because of the smaller similarity score between REALEC development and training sets).
It is interesting to notice that REALEC development and test data sets have a similarity score (i.e. 1.49) significantly lower than FCE development and test data sets (i.e. 3.99). FCE training and development data sets have a similarity score of 1.72. FCE training and test data sets of 3.67. These results might suggest that the English data set is challenging for the models.

Czech (CS) data set – The lower similarity scores between the data sets suggest that systems should perform better on Czech than in English test set. Also if compared to the similarity scores obtained in Italian data sets, the lower similarity scores might indicate that the systems should perform better on Czech than in the Italian test set.

German (DE) data set – Since the low similarity score, indicating a bigger similarity between the sets, should mean that German should be the easiest to tackle for the models.

Italian (IT) data set – Here again, since similarity scores between the sets are lower than in

English one, models should perform better on the Italian data set than in the English. In addition, the higher similarity score between development and test data sets suggests that choosing the best performance model according to the results on the development set should be avoided. Instead training on both training and development data sets should ensure the best performance in this data set.

Swedish (SV) data set – According to the reported similarity scores, Swedish training set is in an order of similarity with development and test sets as the Czech sets. This might suggest that similar performances might be expected.

4 System description

In this section, we describe in detail the specifications of our submission.

Given the nature of the MultiGED shared task, we framed the problem as a token classification task, where systems are required to provide a label for each token within the input sequence. More precisely, we employed a sequence labelling strategy using the BIO labelling schema (Ramshaw and Marcus, 1999). The standard schema is formed by B-I-O tags, where each token in a sentence is labelled with one of the three tags: B indicates the beginning of the error span, i.e. the first token of an incorrect use; I is used to label tokens inside the error unit; O marks tokens that are out of the error span, hence correct. However, since in our task we did not have information about the number of errors nor the error span, we decided to use always B to mark an incorrect token, even when preceded by another incorrect token, and O to mark the correct tokens.

The adopted model allows framing the problem as token classification task that, given a sentence $W = w_1 w_2 \dots w_n$, amounts to labelling each word w_i with B or O tags because of the above-mentioned reason. Figure 2 reports an example of the system output of a sentence from the English FCE training data. Considering the example, we can see that the token *disappointing* is correctly tagged with B, indicating an incorrect usage, and then it is followed by another incorrect token *a*—marked again with the label B because of the information loss from the conversion from error-tagged corpora to binary token labelling. In the same example, the token *week* is labelled as correct while the token *holiday* is labelled as incorrect token.

The model we employed is based on XLM-RoBERTa large: we stacked a linear classifier—with input size of 1024 units and the output size is set to the number of labels—on top of the pre-trained XLM-RoBERTa model, inserting in between the two a dropout layer—with a dropout probability set to 0.1—to avoid overfitting. Finally, in order to compute the distance between the actual data and the predictions we adopted the Cross Entropy loss function. The model architecture is depicted in Figure 3.

To run the experiments, we devised two different experimental settings. In the first one, we trained the models only on the provided training set for 10 epochs, using the development set to select the model achieving the best performance across the training epochs (ELICODE). In the second setting, we trained each model jointly on the training and development sets for 10 epochs, and retained the 10-epoch trained model (ELICODE$_{ALL}$).[5]

To build our models, we started from the ClinicalTransformerNER framework (Yang et al., 2020) and we adapted the code so as to deal with XLM-RoBERTa language model.[6]

Our experiments were performed on machinery provided by the Competence Centre for Scientific Computing (Aldinucci et al., 2017). In particular, we exploited nodes with 2x Intel Xeon Processor E5-2680 v3 and 128GB memory. The training time is about 15 hours per epoch for the provided languages with a large training data—i.e. Czech, English and German—and drops to 8 hours per epoch for Italian and Swedish. The time taken in the prediction phase is about 25 minutes per language.

5 Results and discussion

We report in Table 2 the results obtained by all teams participating to MultiGED shared task (upper part of the table),[7] and additional experimental results—i.e. a baseline and our submitted models but trained in a multilingual fashion (bottom part of the table). As far as the baseline is concerned, we extracted the token counts from the training data and adopted the multinomial Naive Bayes

[5] In both experimental settings we adopted a batch size of 4 and an early stop of 5 epochs.

[6] The code and the models will be publicly available on GitHub after the review phase of this paper to ensure blind review.

[7] We took the results from the official MultiGED repository: https://github.com/spraakbanken/multiged-2023.

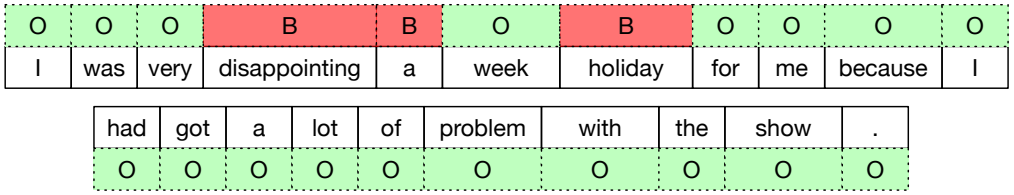


Figure 2: The output of the model for the sentence *I was very disappointing a week holiday for me because I had got a lot of problem with the show.* Here the token *disappointing* is marked as the beginning of an error unit. By the same token, *a* is marked as beginning of a new error due to the information loss caused by the conversion from error-tagged corpora to binary token labelling. The token *holiday* is also marked as an incorrect use. The other tokens are marked as correct uses.

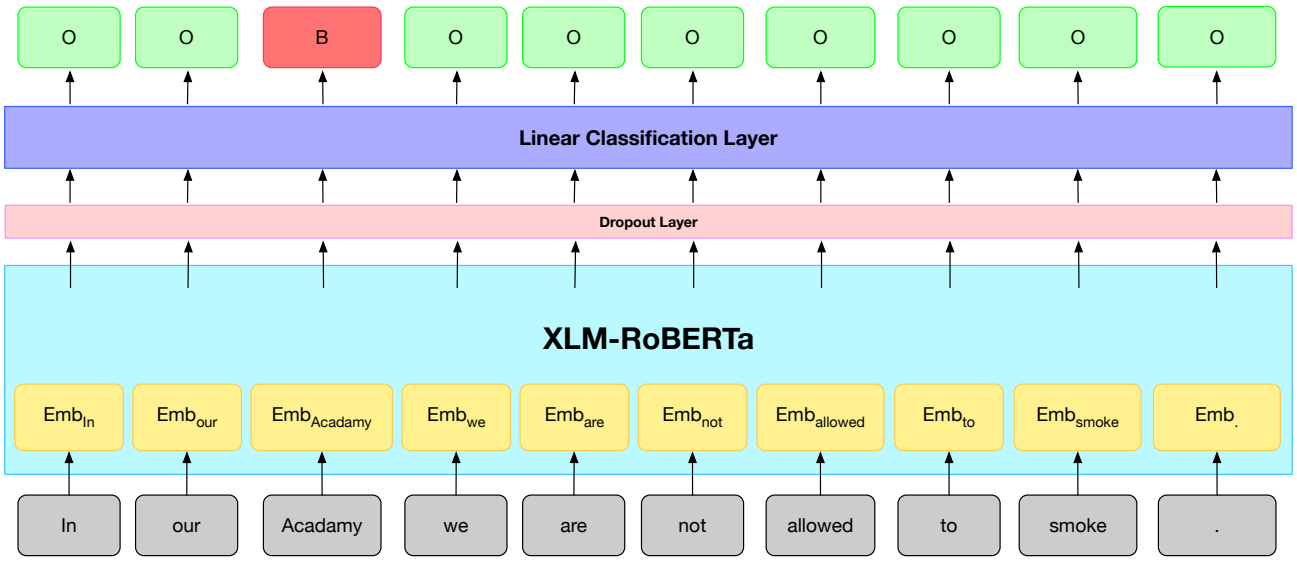


Figure 3: Graphic representation of the model. The grey boxes represent the tokens in the example. These tokens are vectorised and converted into embeddings by XLM-RoBERTa. Tokenisation in XLM-RoBERTa is simplified in this figure for readability reasons. XML-RoBERTa output is inputted to the linear classifier, after passing a dropout layer. The classifier predicts the label B or O for each token.

classifier for sequence labelling (Baseline). As far as the multilingual models are concerned, we followed the same experimental settings of the submitted monolingual models, training two multilingual models: a first model trained only on the concatenation of training data sets (ELICODE$_{MLT}$), the second concatenating also the development data sets (ELICODE$_{MLT_{ALL}}$).

The overall results obtained by both ELICODE and ELICODE$_{ALL}$ are higher than those obtained by the other competing systems, except for the English REALEC test set.

Concerning Precision (P), the baseline and both our ELICODE and ELICODE$_{ALL}$ submissions perform well overall. However, on the FCE partition of the English data set the scores consistently decrease by about 10% and, as expected, the REALEC partition is the most challenging data set: Precision scores drop from about 80% on average to about 40%. As far as Recall (R) is concerned, the token count-based baseline performs poorly: the average Recall of the baseline across languages is about 12% while the average score of ELICODE and ELICODE$_{ALL}$ is about 58%. Following the same trend as Precision, Recall scores for both our submitted systems drop from about 62% of average to 40% on the REALEC English data set. Given the definition of F0.5 metric—i.e. it puts more importance on Precision with respect to Recall—, the overall scores reflect the trend of Precision: the average F0.5 score is about 76% for both ELICODE and ELICODE$_{ALL}$ on all languages but the English REALEC data set, where the average F0.5 drops to 43%.

Considering the different languages, as expected from the quantitative analysis from Section 3, the ELICODE$_{ALL}$ performance improves compared to the scores obtained by ELICODE on Czech, German, Italian and Swedish languages: training on both training and development set allows accounting for the similarities between development set and test set too. Consistently with the above-mentioned analysis, the performances achieved on the Swedish and Czech data sets are comparable and lower than the scores obtained on the German data set, that recorded the highest F0.5 score of 82.32%. Concerning the differences in the English data, as expected, ELICODE performs better than ELICODE$_{ALL}$ on both FCE and REALEC partitions, this is likely due to the high dissimilarity between the English FCE development and test data sets, thus training the model on the development set as well amounts to introducing noise during the learning phase. Additionally, given the great difference between FCE and REALEC partitions, the results of models trained on the FCE data set are consistently lower on REALEC data compared to the results on the FCE data.

In order to explore the impact of the difference between the English data sets, we trained a model only on the REALEC development set. The model has been trained for 10 epochs and by maintaining fixed all the other parameters so as to make the results of such model comparable to the others. The model trained only on REALEC data achieved 58.44 of Precision, 33.19 of Recall and the F0.5 is 50.72, thus improving the F0.5 of about 7% compared to the ELICODE result; in particular, the model becomes more precise in predicting errors, but given the reduced amount of training data is less incline to label tokens as incorrect.

Concerning the baseline, its poor performance is likely due to the employed representation: count-based features consider terms in isolation rather than in context, in so doing, the model is able to detect errors based on words frequency only, thus detecting errors related only to vocabulary—i.e. non-existing words or unseen tokens at training time. In this respect, the results achieved by the baseline on the REALEC partition of the English data set are lower than those for the FCE data set—especially on Precision—, thus reflecting the difference between such two data sets. Conversely, the representations employed by language models such as XLM-RoBERTa are context sensitive—i.e. each token representation accounts for the whole sequence information—and this is reflected in a consistent improvement in Recall scores.

In order to assess the multilingual competence of the language model, we trained a model on the concatenation of the training sets of all the different languages: typologically similar languages may mutually improve the model representations, while languages with different structures may negatively impact the error detection in both languages. The trained multilingual models, as said, follow the same experimental setting than the submitted monolingual models. Differently than the monolingual models which were trained for 10 epochs, the multilingual models have been trained

System	Czech			English - FCE			English - REALEC		
	P	R	F0.5	P	R	F0.5	P	R	F0.5
DSL-MIM-HUS	58.31	55.69	57.76	72.36	37.81	61.18	62.81	28.88	**50.86**
Brainstorm Thinkers	62.35	23.44	46.81	70.21	37.55	59.81	48.19	31.22	43.46
VLP-char	34.93	63.95	38.42	20.76	29.53	22.07	-	-	-
NTNU-TRH	80.65	6.49	24.54	81.37	1.84	8.45	51.34	1.13	5.19
su-dali	-	-	-	-	-	-	-	-	-
ELICODE	82.29	50.61	73.14	73.64	50.34	67.40	44.32	40.73	43.55
ELICODE$_{ALL}$	82.01	51.79	73.44	71.67	50.74	66.21	43.69	40.74	43.07
Baseline	85.69	21.19	53.26	72.81	7.55	26.69	36.40	5.67	17.46
ELICODE$_{MLT}$	83.06	50.72	**73.66**	73.85	50.08	67.45	44.36	42.29	43.93
ELICODE$_{MLT_{ALL}}$	82.79	49.56	73.01	75.01	48.94	**67.79**	45.34	40.29	44.23

System	German			Italian			Swedish		
	P	R	F0.5	P	R	F0.5	P	R	F0.5
DSL-MIM-HUS	77.80	51.92	70.75	75.72	38.67	63.55	74.85	44.92	66.05
Brainstorm Thinkers	77.94	47.55	69.11	70.65	36.46	59.49	73.81	39.94	63.11
VLP-char	25.18	44.27	27.56	25.79	44.24	28.14	26.40	55.00	29.46
NTNU-TRH	83.56	15.58	44.61	93.38	19.84	53.62	80.12	5.09	20.31
su-dali	-	-	-	-	-	-	82.41	27.18	58.60
ELICODE	83.87	71.89	81.16	85.63	66.69	81.03	80.56	67.50	77.56
ELICODE$_{ALL}$	84.78	73.75	**82.32**	86.67	67.96	**82.15**	81.80	66.34	78.16
Baseline	80.99	10.25	34.02	85.11	10.72	35.65	78.09	13.65	40.16
ELICODE$_{MLT}$	83.47	72.52	81.02	85.30	69.64	81.63	82.24	65.94	78.36
ELICODE$_{MLT_{ALL}}$	84.80	71.09	81.65	85.71	65.95	80.87	83.34	64.37	**78.70**

Table 2: Results of experiments in the token classification task. To increase readability, we partitioned the results on two tables grouped by language. We reported the results for all the systems submitted to the MultiGED competition—in the upper part of each sub-table—together with the results of our submission (ELICODE and ELICODE$_{ALL}$). The bottom part of each sub-table report the Naive Bayes-based baseline and the multilingual models (ELICODE$_{MLT}$ and ELICODE$_{MLT_{ALL}}$) results. For each system we report the scores obtained on all the languages included in the competition; for each language, the corresponding columns report the Precision (P), Recall (R) and F0.5 scores. The highest F0.5 scores are in bold.

for 7 epochs: in this setting the training took on average 55 hours per epoch for ELICODE$_{MLT}$ and 62 hours for ELICODE$_{MLT_{ALL}}$.[8]

The multilingual models perform similarly on the shared task test sets compared to monolingual models. If we consider the two languages with a smaller training and development sets, i.e. Italian and Swedish, we might notice that the performance on the Italian test set does not improve using the multilingual approach. This might be due to the fact that the other languages included in the shared task are not typologically similar to Italian. On the contrary, the performance on the Swedish language, which is slightly higher than the monolingual model performance, might benefit from the German training and development data sets, being both Germanic languages.

6 Conclusion and future work

In this paper, we presented the ELICODE system submitted to the first shared task on Multilingual Grammatical Error Detection (MultiGED). We studied the effect of fine-tuning the pre-trained XLM-RoBERTa language model on the multilingual grammatical error detection framed as sequence labelling task. The submitted system achieved the highest scores on five out of six different data sets in a multilingual setting: the provided data are in five languages, namely Czech, English, German, Italian and Swedish.

We compared our system with a simple Naive Bayes classifier based on token counting. The comparison shows that a system based on local representations is able to detect a small subset of errors (good Precision and low Recall) such as typos or out-of-vocabulary words; conversely, a system exploiting contextual representations detects a larger number of error types (increased Recall). Additionally, we compared our monolingual system with a multilingual model trained jointly on the five-language training data sets. We found that the results achieved by the multilingual model are comparable to those obtained by the monolingual models, thus indicating that the token representations built by the language model are suited to generalise over different languages.

As part of future work, we plan to qualitatively analyse the error types recognised by the presented models, to find possible ways to improve grammatical error detection, e.g. by creating hybrid or ensemble models, but also to verify that models based on local representations are able to recognise mainly error categories based on the *signifier*, which do not need to take context into account. Another interesting solution could be that described in Omelianchuk et al. (2020), in which the authors address the GEC task iteratively.

Concerning error types and interlanguage, it would be interesting to train Second Language Acquisition theory-aware models taking interlanguage stages into account by grouping data according to CEFR level information. Indeed, learners at the same learning stage share the same error types, irrespective to their mother tongue (Giacalone Ramat, 2003). These models might perform better in applicative cases in which we know learners' language level (Bryant et al., 2019).

In addition, it would be interesting to analyse the embeddings generated by models fine-tuned on this task, using visualisation techniques as principal component analysis, to verify if embeddings representing the same word are localised in different space areas according to their correct or incorrect usage.

Furthermore, we plan to explore the performance of other language models already tested in GEC and GED tasks to compare RoBERTa and other transformer-based models trained using a different technique (e.g. ELECTRA trained to discriminate the wrongly generated token in a sequence).

[8] The multilingual model trained only on the training data sets (ELICODE$_{MLT}$) for 7 epochs achieved the same results of the 8-epoch model. Thus, we assume that ELICODE$_{MLT}$ reached the learning upper bound at the 7th epoch.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.

M Aldinucci, S Bagnasco, S Lusso, P Pasteris, S Rabellino, and S Vallero. 2017. OCCAM: a flexible, multi-purpose and extendable HPC cluster. *Journal of Physics: Conference Series*, 898(8):082039.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Samuel Bell, Helen Yannakoudakis, and Marek Rei. 2019. Context is key: Grammatical error detection with contextual word representations. In *Proceedings of the Fourteenth Workshop on Innova-*

tive Use of NLP for Building Educational Applications, pages 103–115, Florence, Italy. Association for Computational Linguistics.

Adriane Boyd. 2018. Using wikipedia edits in low resource grammatical error correction. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 79–84.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2022. Grammatical Error Correction: A Survey of the State of the Art. *arXiv preprint arXiv:2211.05166*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Elisa Di Nuovo. 2022. *VALICO-UD: annotating an Italian learner corpus*. Doctoral Thesis. University of Genoa and University of Turin.

Jonas Gehring, Michael Auli, David Grangier, and Yann N Dauphin. 2016. A convolutional encoder model for neural machine translation. *arXiv preprint arXiv:1611.02344*.

Anna Giacalone Ramat. 2003. *Verso l'italiano. Percorsi e strategie di acquisizione*. Roma, Carocci.

Sylviane Granger. 1998. The computer learner corpus: a versatile new source of data for SLA research. In *Learner English on computer*, pages 3–18. Routledge.

Sylviane Granger, Maïté Dupont, Fanny Meunier, and Magali Paquot. 2020. International Corpus of Learner English. Version 3 (Handbook and web interface). *Louvain-la-Neuve: Presses Universitaires de Louvain*, page 134.

Carl James. 1998. *Errors in language learning and use*. Pearson Educational Limited.

Masahiro Kaneko and Mamoru Komachi. 2019. Multi-head multi-layer attention to deep language representations for grammatical error detection. *Computación y Sistemas*, 23(3):883–891.

Adam Kilgarriff. 2001. Comparing corpora. *International journal of corpus linguistics*, 6(1):97–133.

Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlỳ, and Vít Suchomel. 2014. The sketch engine: ten years on. *Lexicography*, 1(1):7–36.

Elizaveta Kuzmenko and Andrey Kutuzov. 2014. Russian error-annotated learner English corpus: a tool for computer-assisted language learning. In *Proceedings of the third workshop on NLP for computer-assisted language learning*, pages 87–97, Uppsala, Sweden. LiU Electronic Press.

Paul Lennon. 1991. Error: Some problems of definition, identification, and distinction. *Applied linguistics*, 12(2):180–196.

David Little. 2006. The Common European Framework of Reference for Languages: Content, purpose, origin, reception and impact. *Language Teaching*, 39(3):167–190.

Anke Lüdeling and Hagen Hirschmann. 2015. Error annotation systems. In Sylviane Granger, Gaëtanelle Gilquin, and Fanny Meunier, editors, *The Cambridge handbook of learner corpus research*, pages 135–157. Cambridge University Press, Cambridge.

Jakub Náplava, Milan Straka, Jana Straková, and Alexandr Rosen. 2022. Czech grammar error correction with a large and diverse corpus. *Transactions of the Association for Computational Linguistics*, 10:452–467.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The conll-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14.

Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. 2013. The CoNLL-2013 shared task on grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12, Sofia, Bulgaria. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR – grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170, Seattle, WA, USA → Online. Association for Computational Linguistics.

Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1756–1765, Vancouver, Canada. Association for Computational Linguistics.

Matthew E Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. Dissecting contextual word embeddings: Architecture and representation. In *EMNLP, Association for Computational Linguistics*, pages 1499–1509.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Marek Rei. 2017. Semi-supervised multitask learning for sequence labeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 2121–2130.

Larry Selinker. 1972. Interlanguage. *International Review of Applied Linguistics*, 10(3):209–231.

Marco Siino, Elisa Di Nuovo, Ilenia Tinnirello, and Marco La Cascia. 2022. Fake news spreaders detection: Sometimes attention is not all you need. *Information*, 13(9):426.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Elena Volodina, Christopher Bryant, Andrew Caines, Orphée De Clercq, Jennifer-Carmen Frey, Elizaveta Ershova, Alexandr Rosen, and Olga Vinogradova. 2023. MultiGED-2023 shared task at NLP4CALL: Multilingual Grammatical Error Detection. In *Proceedings of the 12th workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL)*, pages 1–15, Tórshavn, Faroe Islands.

Elena Volodina, Lena Granstedt, Arild Matsson, Beáta Megyesi, Ildikó Pilán, Julia Prentice, Dan Rosén, Lisa Rudebeck, Carl-Johan Schenström, Gunlög Sundberg, et al. 2019. The swell language learner corpus: From design to annotation. *Northern European Journal of Language Technology (NEJLT)*, 6:67–104.

Xi Yang, Jiang Bian, William R Hogan, and Yonghui Wu. 2020. Clinical concept extraction using transformers. *Journal of the American Medical Informatics Association*. Ocaa189.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Helen Yannakoudakis, Øistein E Andersen, Ardeshir Geranpayeh, Ted Briscoe, and Diane Nicholls. 2018. Developing an automated writing placement system for ESL learners. *Applied Measurement in Education*, 31(3):251–267.

Zheng Yuan, Shiva Taslimipoor, Christopher Davis, and Christopher Bryant. 2021. Multi-class grammatical error detection for correction: A tale of two systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8722–8736, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.